Homework lecture 1
Introduction to Distributed Algorithms

Jaap-Henk Hoepman
jhh@cs.ru.nl
February 5, 2018

Question 1: Will the following program (eventually) terminate? Assume that reading or writing a single variable is atomic.

\[ i ← 0 \]
\[ j ← 0 \]

\textbf{thread while} \( i = 0 \)
\hspace{1em} \textbf{DO} \( j ← j + 1 \mod 2 \); print \( i \)
\hspace{1em} print \( i \)

\textbf{thread while} \( i = 0 \)
\hspace{1em} \textbf{do if} \( j = 0 \) \textbf{then} \( i = 1 \)

Question 2: Will the following program (eventually) terminate, or is it possible that it runs forever? Assume that reading or writing a single variable is atomic.

\[ a ← 1 \]
\[ b ← 1 \]

\textbf{thread while} \( a ≠ 0 \)
\hspace{1em} \textbf{do} \( b ← (b + a) \mod 2 \)

\textbf{thread while} \( b ≠ 0 \)
\hspace{1em} \textbf{do} \( a ← a + 1 \)
\hspace{1em} \( a ← 0 \)

Question 3: Lamport’s logical clock algorithm works in the message passing model. Modify Lamport’s logical clock algorithm to assign logical clock values to all events in a shared memory system that supports atomic reads and atomic writes to shared memory. Prove that the logical clock created by your algorithm can be used to put the events in a total order \( \langle A, ⇒ \rangle \) consistent with the partial order \( \langle A, → \rangle \).